Ports, Protocols, and the OSI Model

Cybersecurity

Networking 101, Day 2

Class Objectives

By the end of today's class, you will be able to:

- Interpret data in network packets by analyzing their headers, payloads, and trailers.
- Explain the role of ports in specifying a network packet's destination.
- Associate common protocols with their assigned ports.
- Explain how encapsulation and decapsulation allow different protocols to interact with one another.
- Use the layers of the OSI model to identify the source of problems on a network.
- Capture and analyze live network traffic using Wireshark.

Protocols

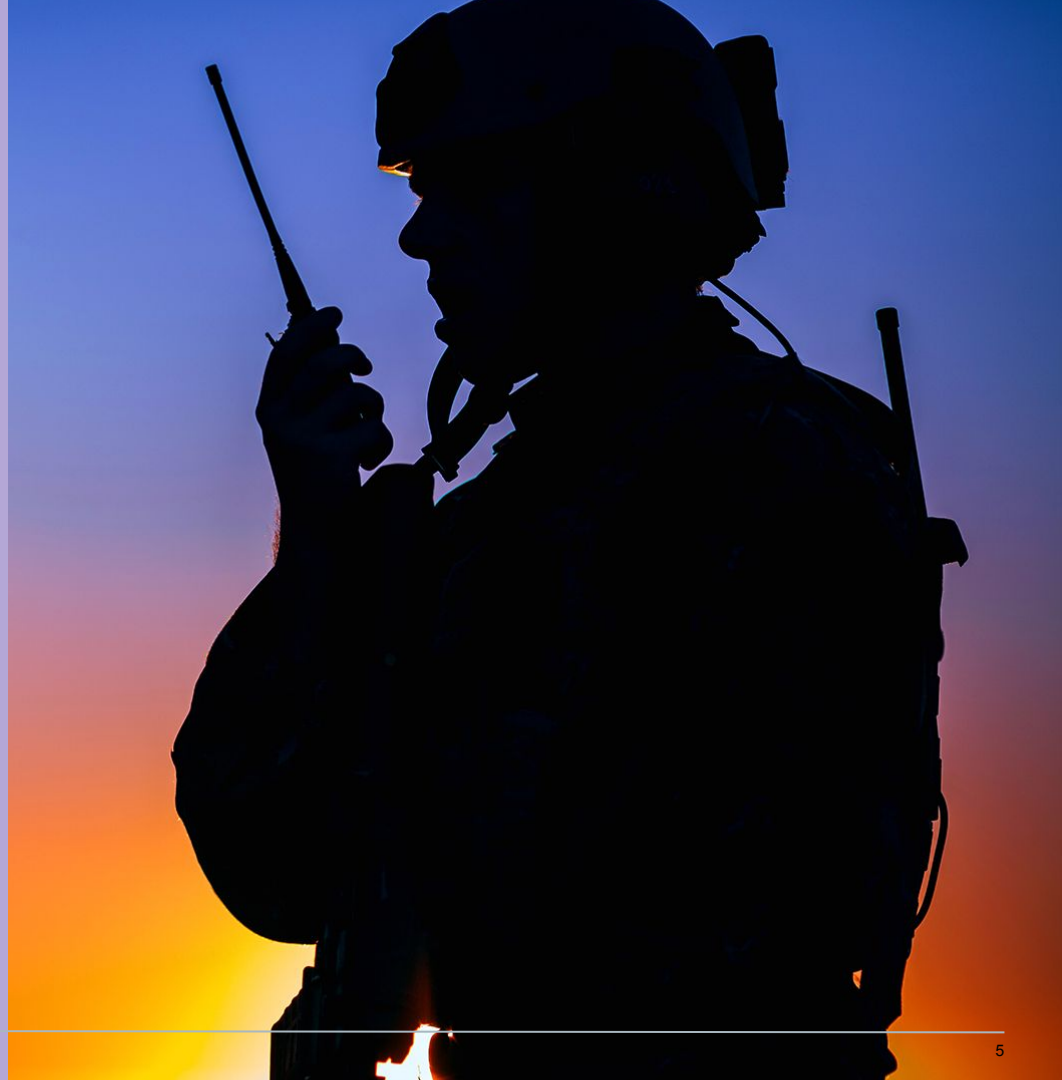
“Roger, Over, and Out”


With so much
going on behind-the-scenes
in network communication,
it’s important to have systems
in place to avoid the many
potential errors.

Real World Protocols

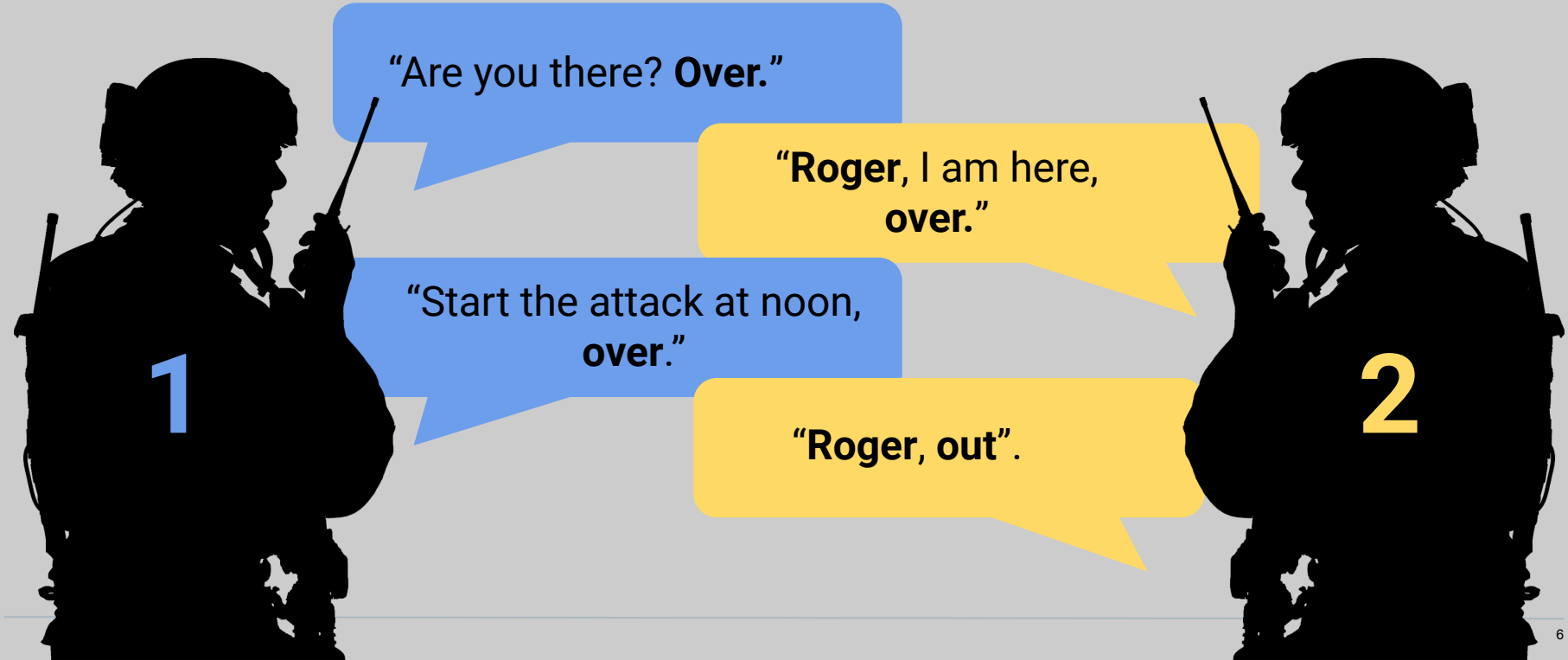
When military personnel need to exchange messages over a radio, they may encounter the following issues:

- Communications don't come in clearly.
- Multiple communications are sent at the same time.
- Communications are cut off.



Real World Protocols

Military personnel use **communication protocols** to ensure messages are *transmitted*, *received*, and *understood* clearly.



Real World Protocols

This mode of communication ensures that there is no ambiguity when sending and receiving messages. Clear meanings for certain keywords and strict rules for using them drastically improves the efficacy of communications.

- “**Over**” signals the end of a specific line of communication.
- “**Roger**” signals that the message was received completely.
- “**Out**” signals that the message was received, the exchange is complete, and the order will be followed.

These strict rules are known as a **protocol**. They impose structure by specifying the precise meaning of keywords, and where in a message they must appear.

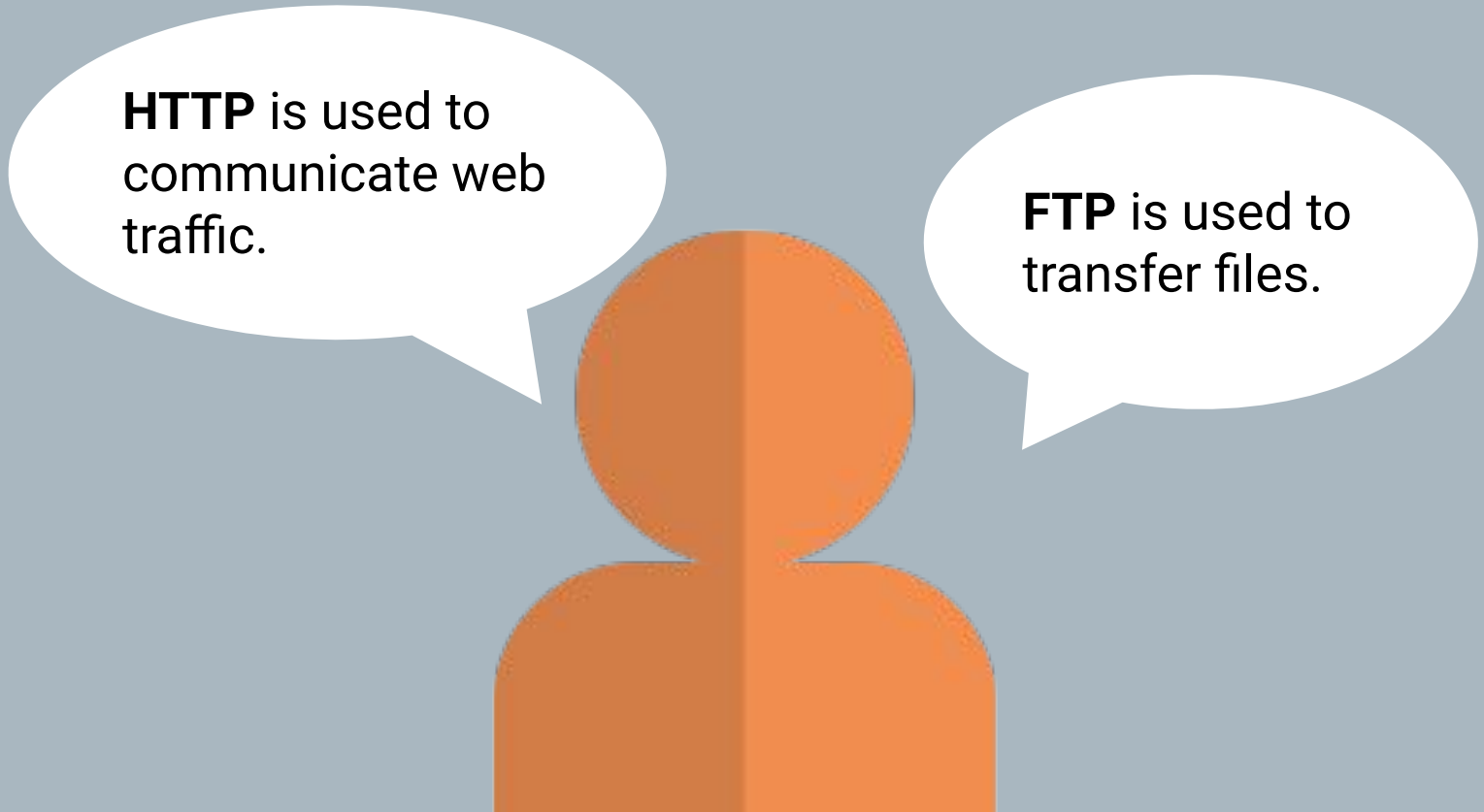
Protocols and Networks

Networks use protocols to ensure messages are fully sent and understood.

Similar to the military's use of "**over**," a network uses the TCP message **`FIN`** to indicate the end of the transmission.

Networking Protocols

Some common protocols you may be familiar with:



Networking Protocols

Some other important protocols:

PAP is used for authenticating a user.

SMB is a Windows-based protocol for sharing files.

NetBIOS allows computers to communicate on a local network.

As we know
from the previous class,
the client and server
communicate by
exchanging
binary data.

Network Packets

Packets

The binary data is grouped together into separate pieces, known as **packets**, and transmitted across the network .

Header	Sender's IP address, receiver's IP address, protocol	96 bits
Payload	Data	660 bits
Trailer	Indicates end of packet, error correction	32 bits

The specific arrangement of packets allows the receiving party to properly interpret the contents and direction of the communication.

Packets

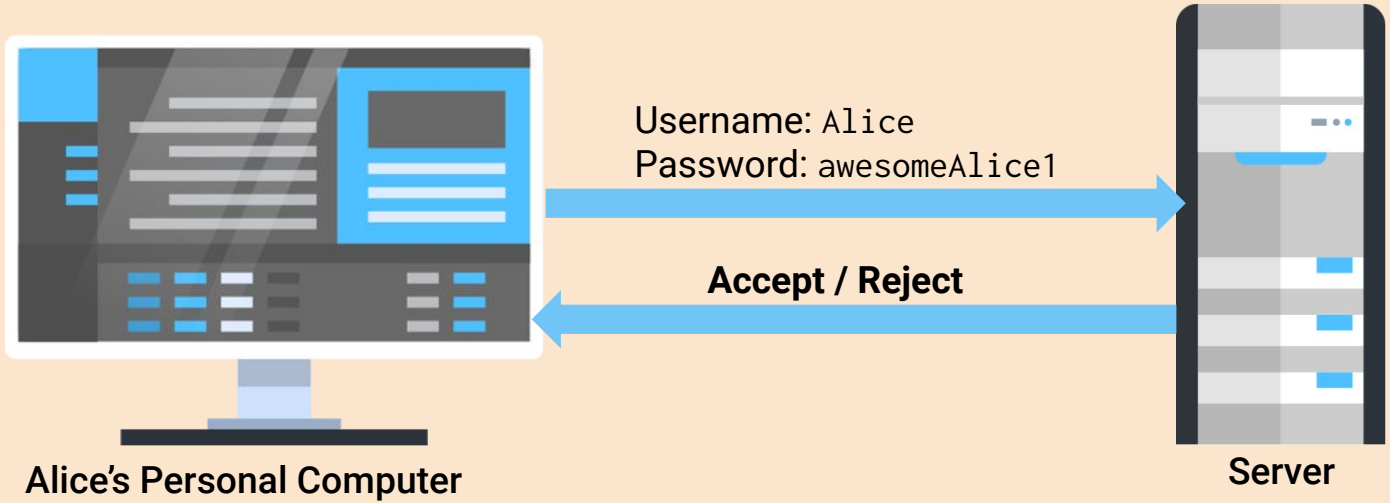
For example, the **version** field is indicated in the header. As the first field, it starts at the first bit and ends at the fourth bit. The receiver will always find this information in this exact location.

010001010000000000000001110110100000111100011000100000000000000100000000000110

In binary, 0100 translates to 4. This field indicates that this header uses Version 4.

Protocol Example: Authentication

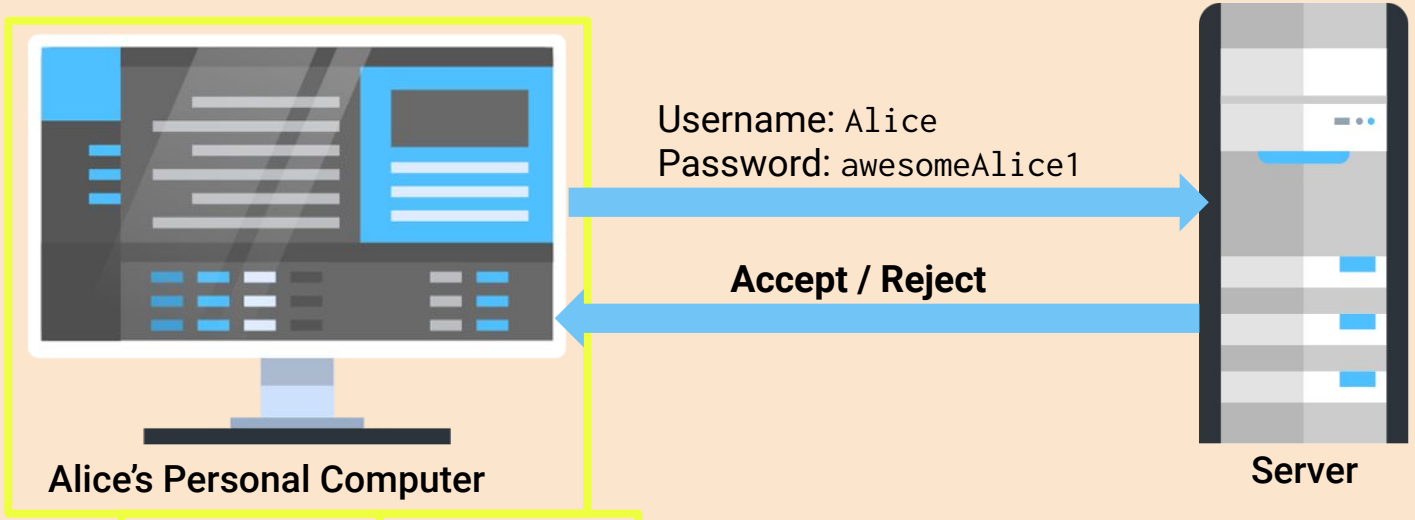
PAP (*Password Authentication Protocol*)



PAP Two-Way Handshake

Protocol Example: Authentication

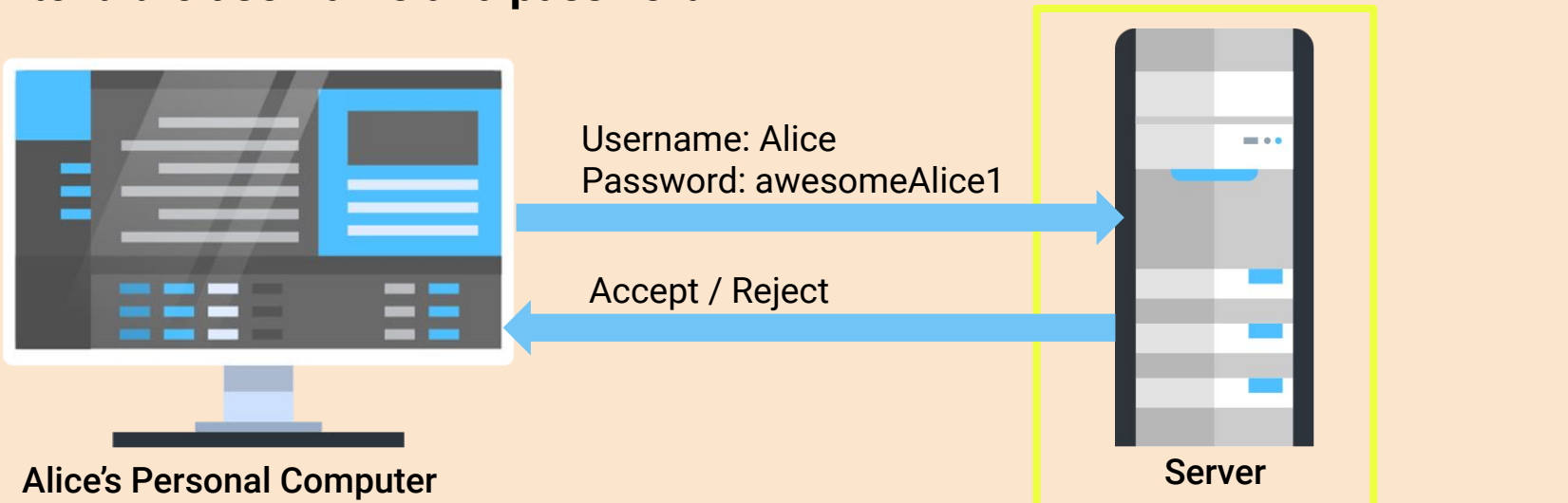
Client request contains bits in a specific **order** and **length**, per the standard and rules of the protocol.



1 Byte	1	2 Bytes	1 Byte	Variable	1 Byte	Variable
Code: 1	ID	Length	User name length	Username	Password Length	Password

Protocol Example: Authentication

The server receiving the request will know where to look in the bitstream for content: the **username** and **password**.

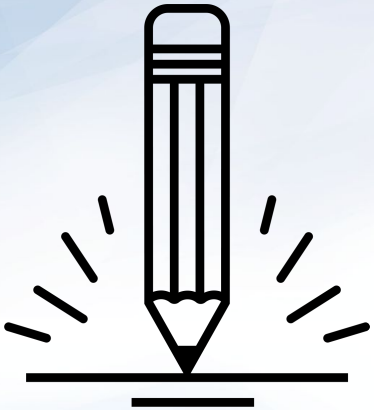


1 Byte	1	2 Bytes	1 Byte	Variable	1 Byte	Variable
Code: 1	ID	Length	User name length	Username	Password Length	Password

Interpreting Protocols from Raw Binary Data

Security professionals rely on web tools to convert binary data and determine the protocol being used.

```
01110000011000010111000000100000
01110011011001010110111001110100
00101110101110101011100110110010 1
01110010011011100110000101101101
01100101001000000101000001000001
01010000010101010100110100010 1
01010010001000000111000001100001
01110011011100110111011101101111
01110010011001000010000000110111
```



Activity: Interpreting Protocols

In this activity, you will continue to play the role of a security analyst at Acme Corp.

You must convert raw binary data into a readable format and determine which protocol is being used.

Suggested Time:
15 Minutes

Time’s Up! Let’s Review.

Ports

Ports Real World Analogy

- Bob wants to show a video presentation to Alice, a coworker who works across town. Bob will be presenting from Room 33, the video conference room in his office building.
- Bob told Alice to meet him at his office, but only gave her the building address, 150 Main Street.
- Because Bob didn't specify which room he'd be in, Alice will be able to find the office building, but not Bob or the video presentation.

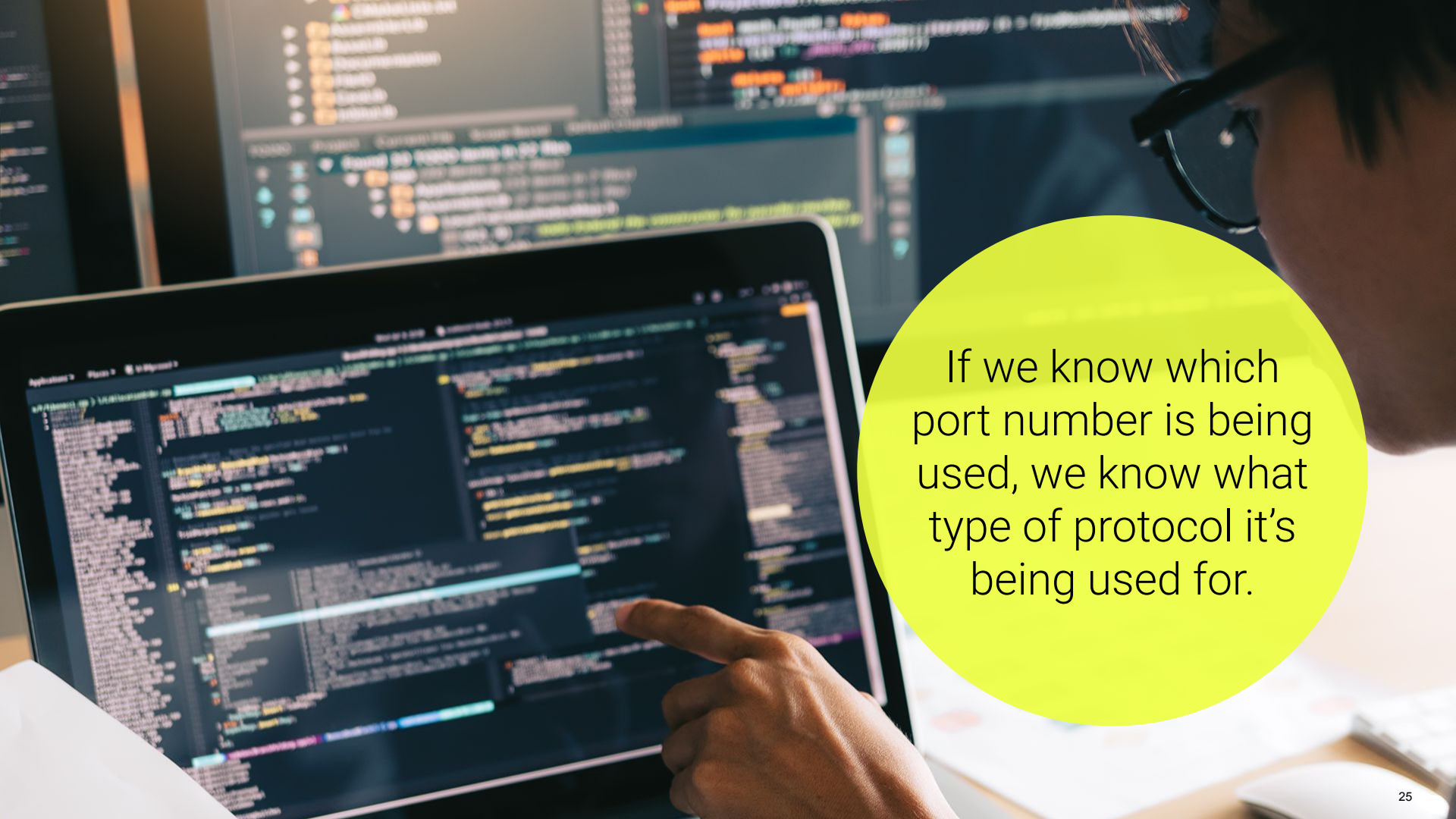
Ports Real World Analogy

In this analogy, the address of the office building, 150 Main Street, is the IP address.

The video conference room, Room 33, is the port.

Since we know that Room 33 is the video conference room, we know that any meeting in Room 33 will involve video conferencing, even if we're not explicitly told what to expect from the meeting.

Similarly, port numbers can be associated with a specific network function and protocol.


If we know which port number is being used, we know what type of protocol it’s being used for.

Ports and Security

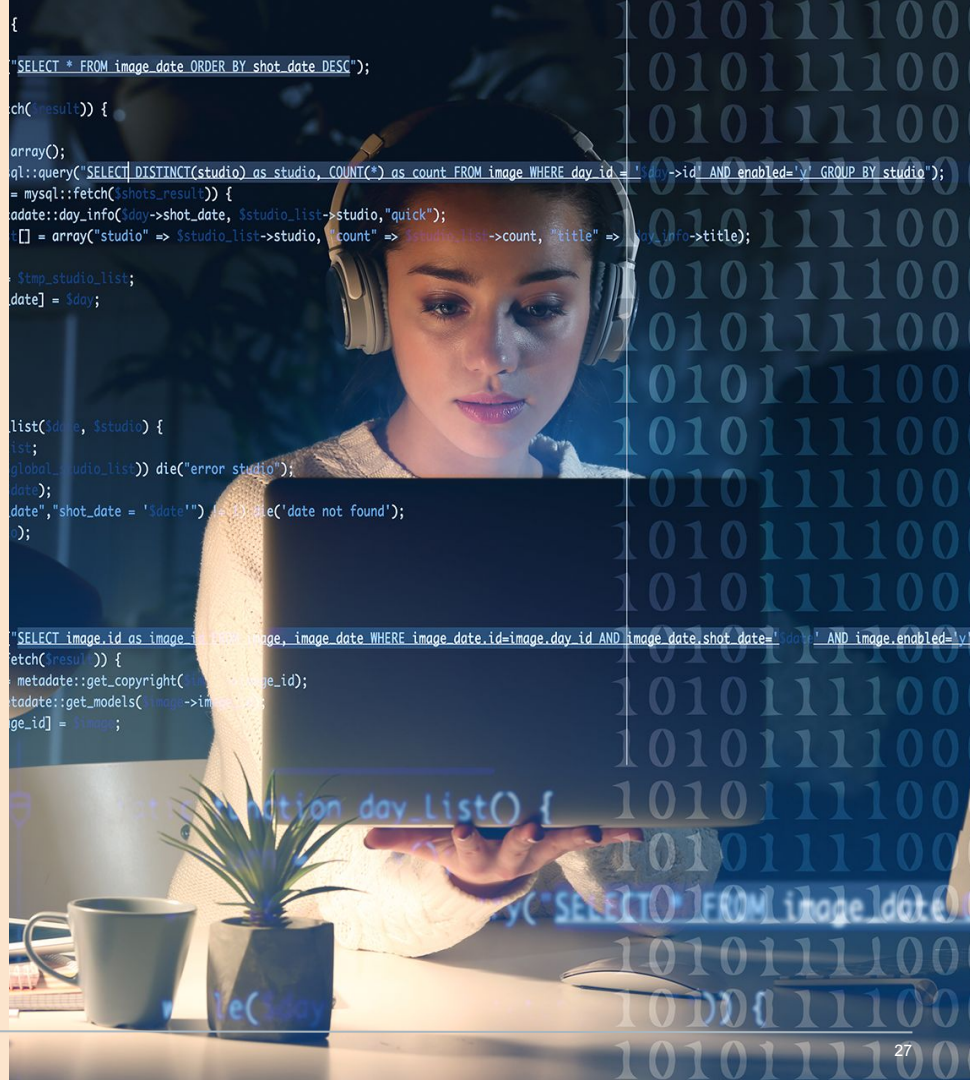
Ports are the access points for transmitting and receiving data.

- Ports are like doors that can be opened, closed, or accessible only to certain individuals.
- It’s important that IT professionals do not allow unauthorized access to these points of entry. Unauthorized access can potentially lead to a breach.

Virtual Ports

Computers don’t have enough physical space for every protocol, so we use software to create **virtual ports**.

- Every protocol is assigned a numerical virtual port number.
- The corresponding port is the **destination port**. It’s where other machines send data to communicate with that protocol.
    - For example: A machine sending an HTTP message to a web server sends traffic to the server’s port `80`.



Port Numbers

These ports are divided into three ranges:

There are 65,536 virtual ports, numbered from `0` to `65535`.

01 System Ports

02 Registered Ports

03 Dynamic Ports

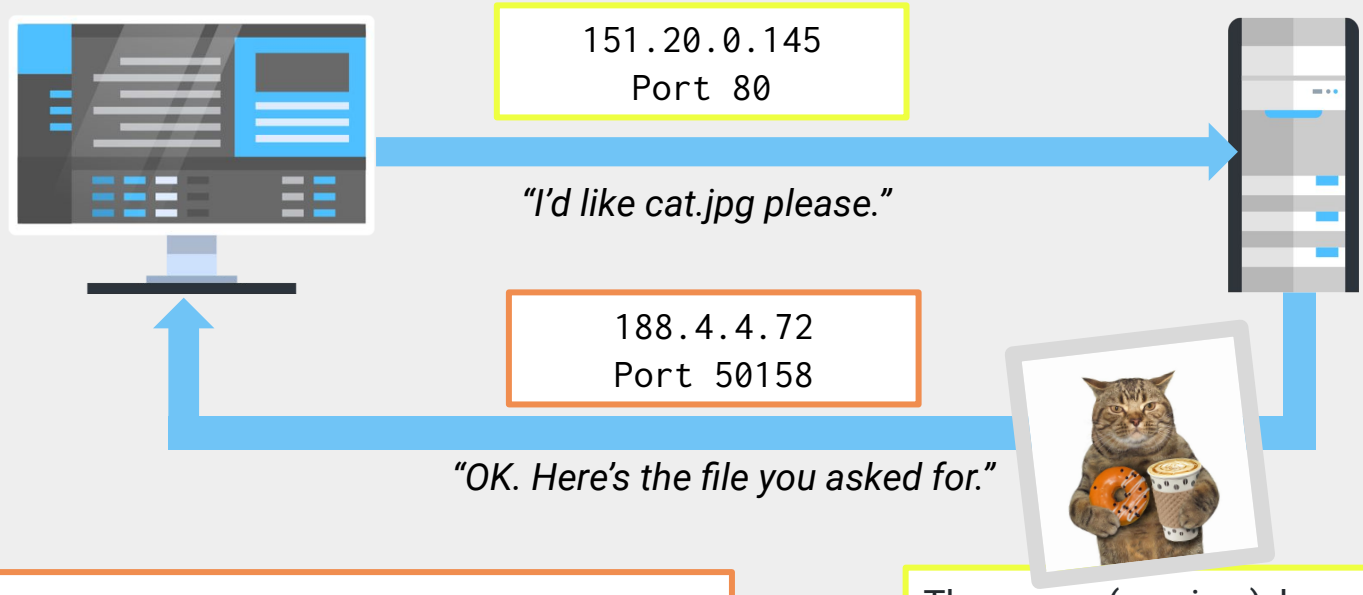
Port Numbers

Range Title	Range Number	Also Known As	General Explanation
System Ports	**Range** `0-1023`	Well-known ports	Restricted: Only the operating system or administrators can bind services to these ports. E.g., HTTP typically runs on port `80`. Normal users can't launch services on port `80`, so we can trust that machines accepting connections to port `80` are using it to send and receive HTTP traffic.
Registered Ports	**Range** `1024-49151`	User ports	“Normal” users launching their own services will do so using ports in this range.
Dynamic Ports	**Range** `49152-65535`	Dynamic ports/private ports	When a machine sends data to another machine, it must open a port to send from. This is called a source port. Source ports are randomly chosen from the dynamic range whenever a machine sends a message.

Common Ports

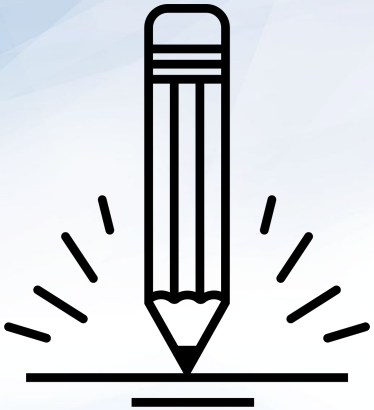
Port 80	**HTTP**	Sending web traffic.
Port 443	**HTTPS**	Sending encrypted web traffic.
Port 21	**FTP**	Sending files.
Port 22	**SSH**	Securely operating network services.
Port 25	**SMTP**	Sending emails.
Port 53	**DNS**	Translating domains into IP addresses.

Source / Destination Ports



The client (initiator) has a **source** port. Source ports are randomly generated from the dynamic port range (49152 - 65536).

The server (receiver), has a **destination** port, depending on the protocol. Destination ports don't change, so we can associate a port with a certain protocol.



Activity: Ports

In this activity, you will continue to play the role of a security analyst at Acme Corp.

You will find the source and destination port for several network requests and determine the protocol for each destination port.

Suggested Time:
15 Minutes

Time’s Up! Let’s Review.

Countdown timer

15:00

(with alarm)

OSI Layers

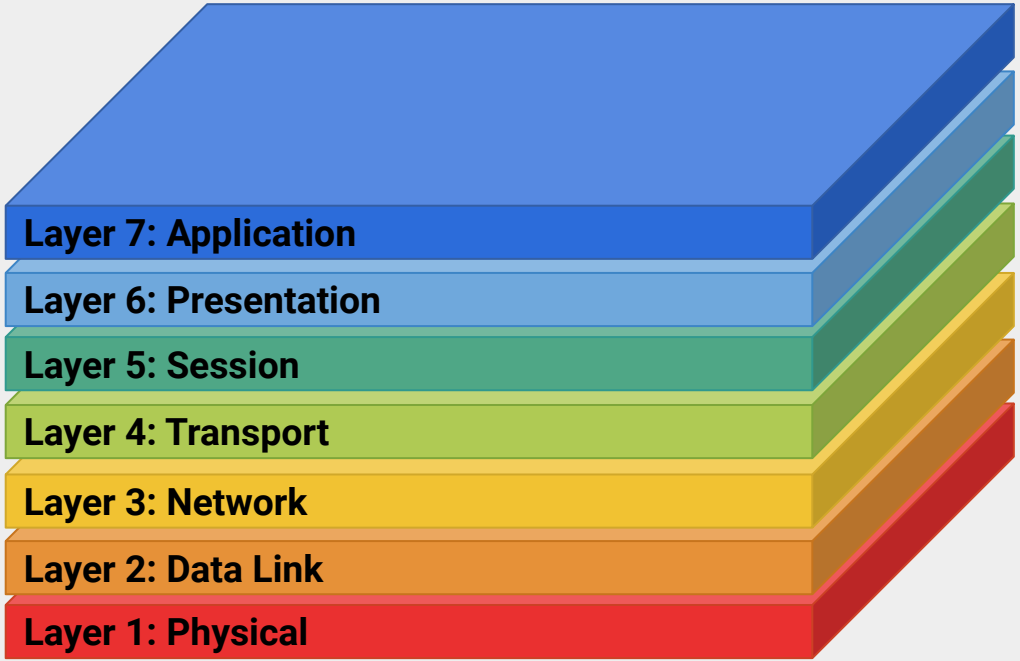

When data travels across a network, it goes through multiple steps and processes to reach its destination.

OSI Layers

	The process of sending an email starts with the following steps:	Certain protocols are responsible for handling specific steps of data transmission:
1.	Convert the text of the user's email into a format the email application can understand.	The IMAP or POP3 converts the text of a user's email into a format any email application can understand.
2.	Add the destination address and destination port to this data.	The TCP adds information about destination ports to this data.
3.	Convert this packet to a format that can be transmitted through physical wires.	The IP adds destination address information to the email data.

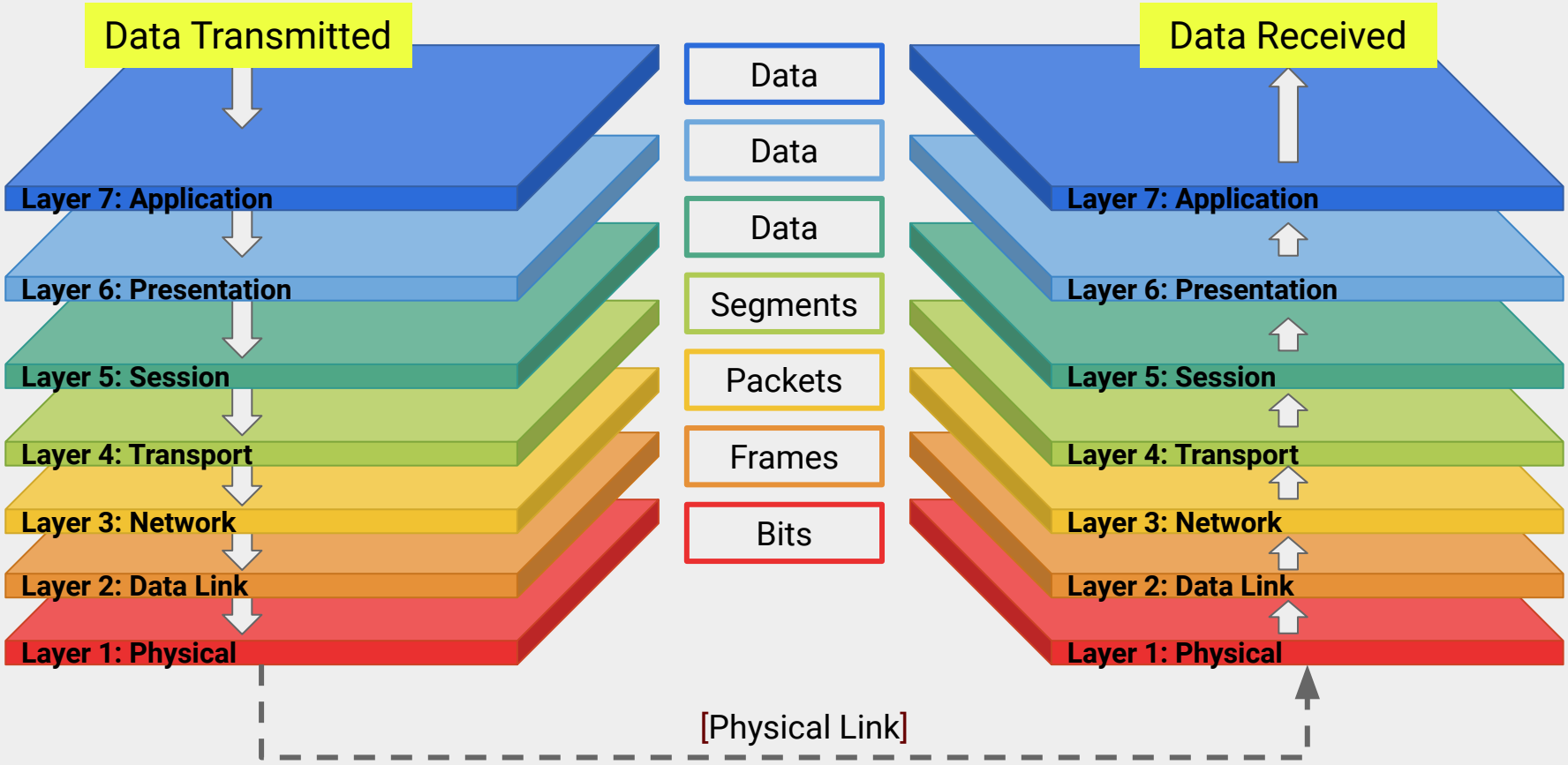
The Open Systems Interconnection (OSI) Model

The OSI model provides a framework for categorizing and conceptualizing the large number of ports and protocols.



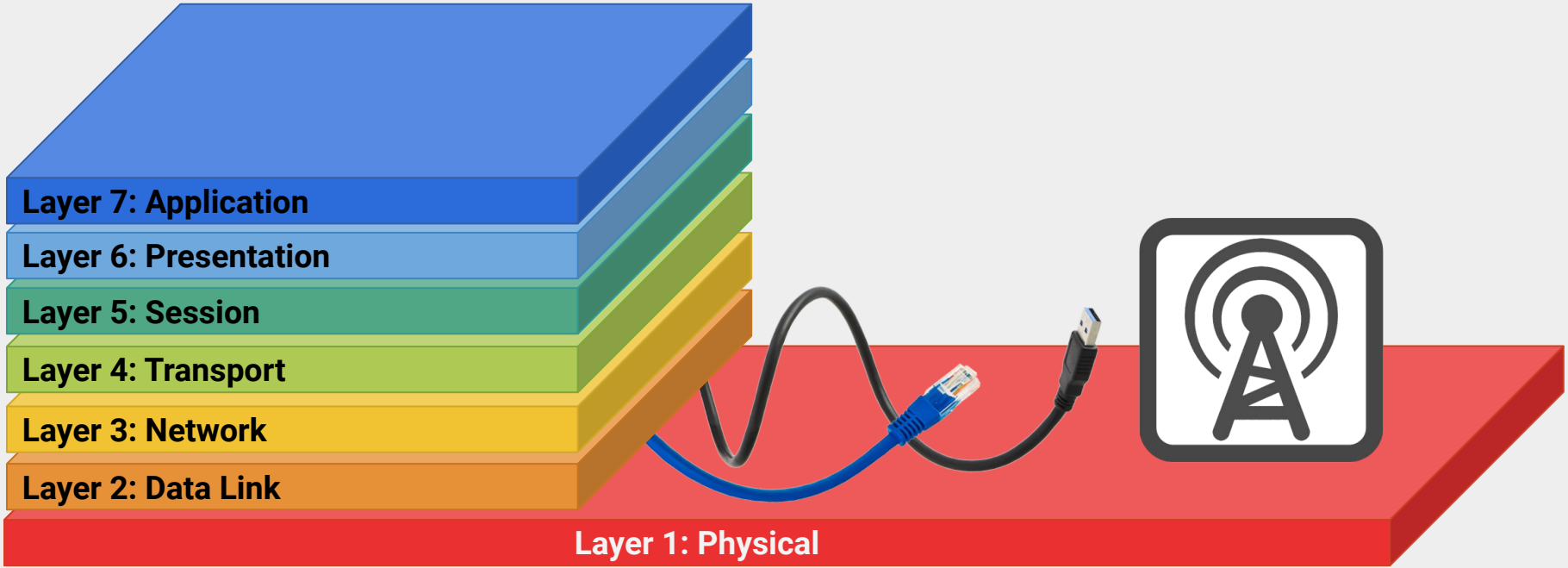
The OSI model is a seven layer framework that allows security analysts to understand how communication works on a network, by detailing the processes, devices, and protocols in place at each layer.

OSI Model


Data Transmitted
Data Received
Layer 7: Application
Layer 6: Presentation
Layer 5: Session
Layer 4: Transport
Layer 3: Network
Layer 2: Data Link
Layer 1: Physical
Data
Data
Data
Segments
Packets
Frames
Bits
[Physical Link]

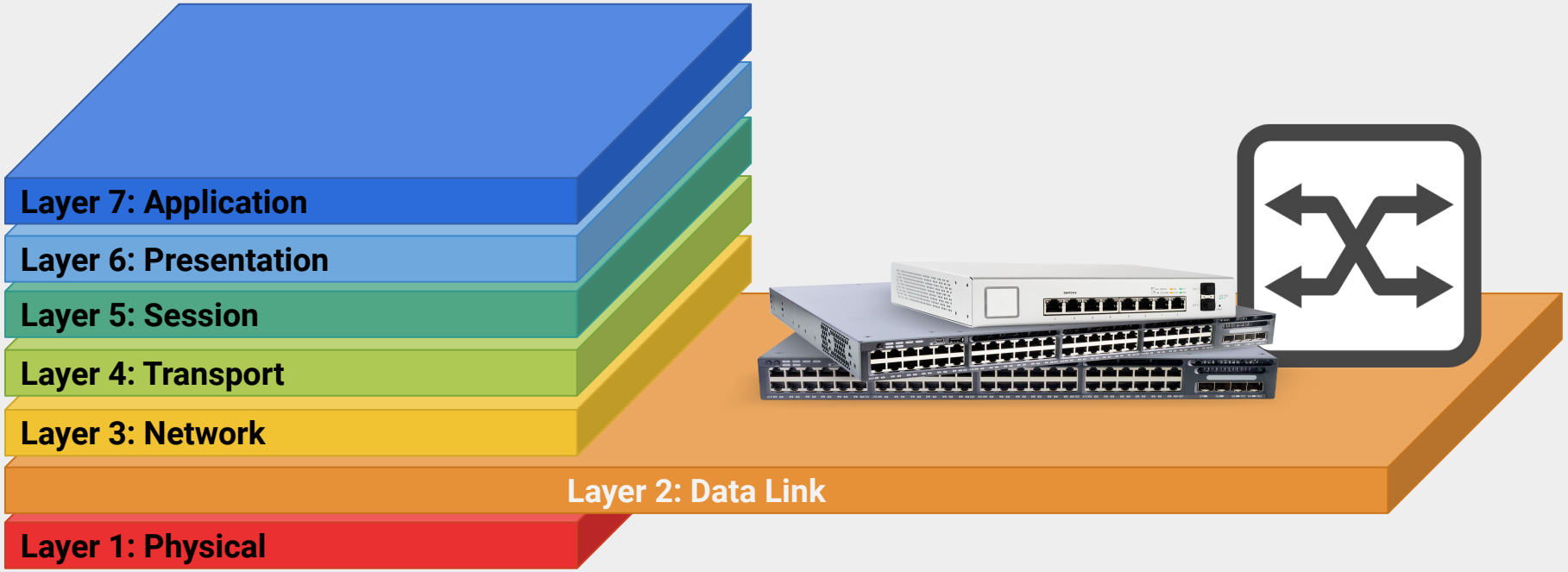
Layer 1: Physical

The **Physical layer** is responsible for transmission of binary data through a physical medium. It handles how data is physically encoded and decoded.



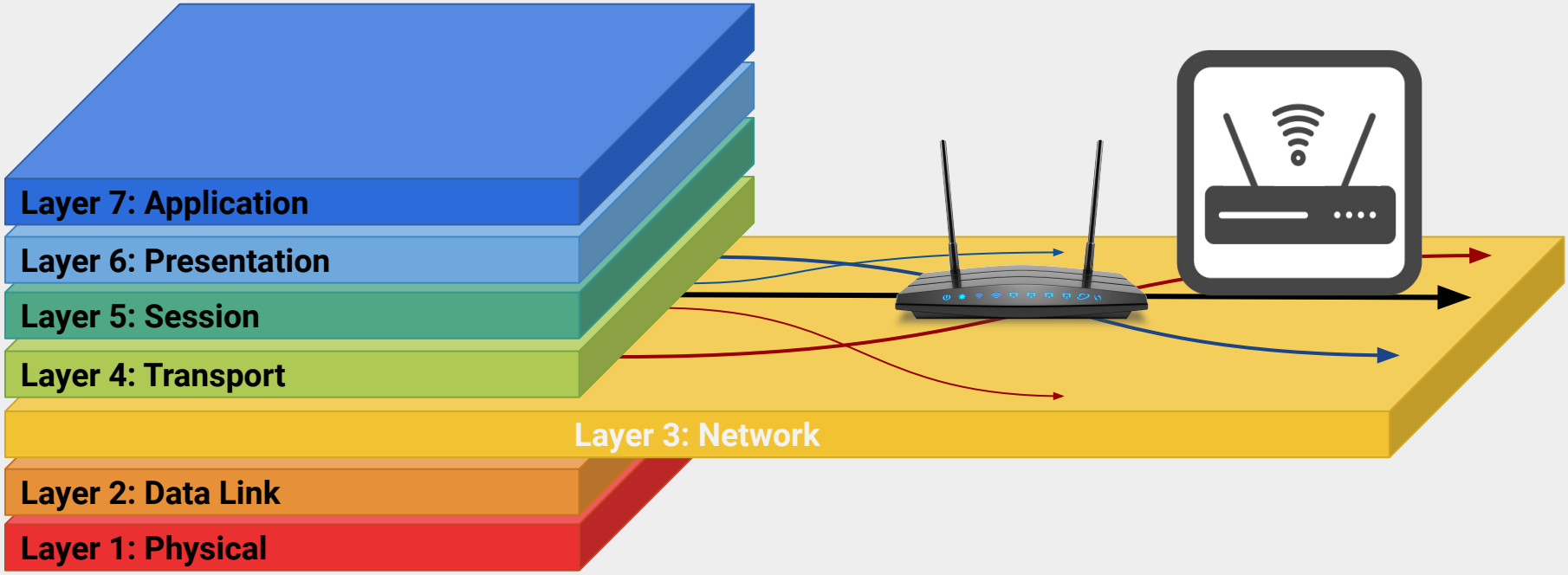
Layer 2: Data Link

The **Data Link layer** establishes links between nodes. It also ensures data gets to its final destination without corruption, thus protecting data integrity.



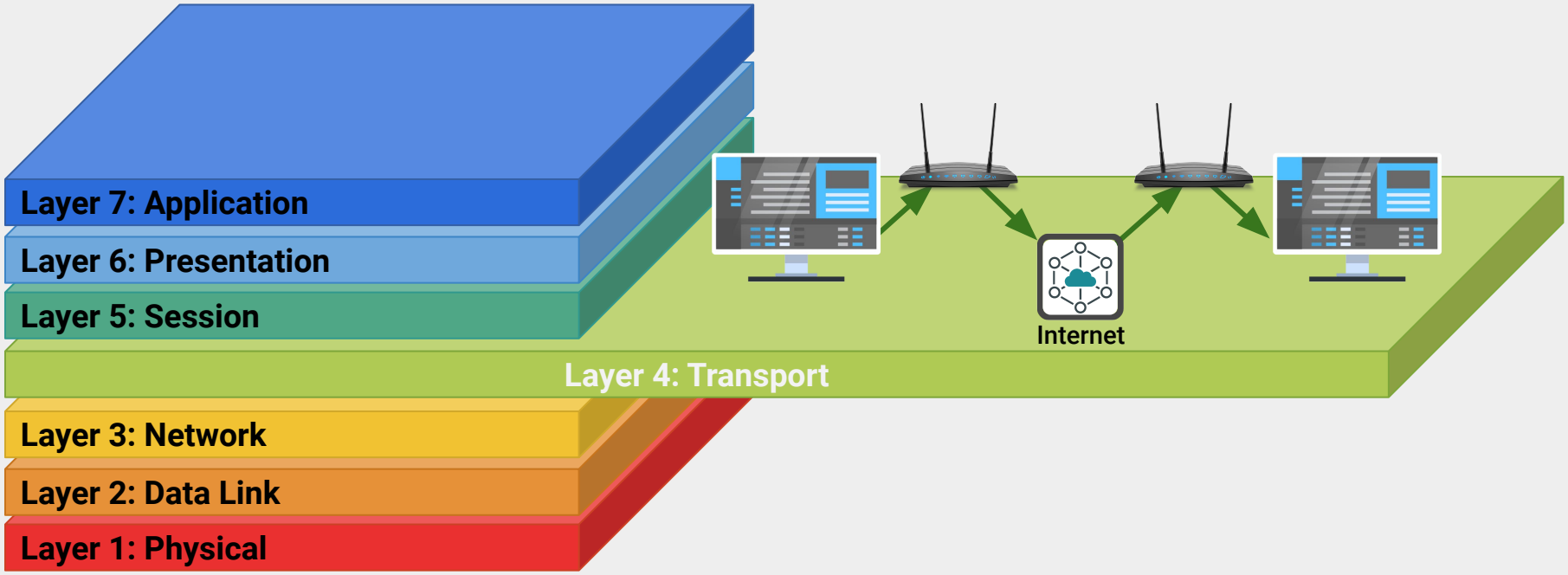
Layer 3: Network

The **Network layer** routes data through physical networks using an IP address, deciding which physical path the data will take, and ensuring it gets to the correct destination.



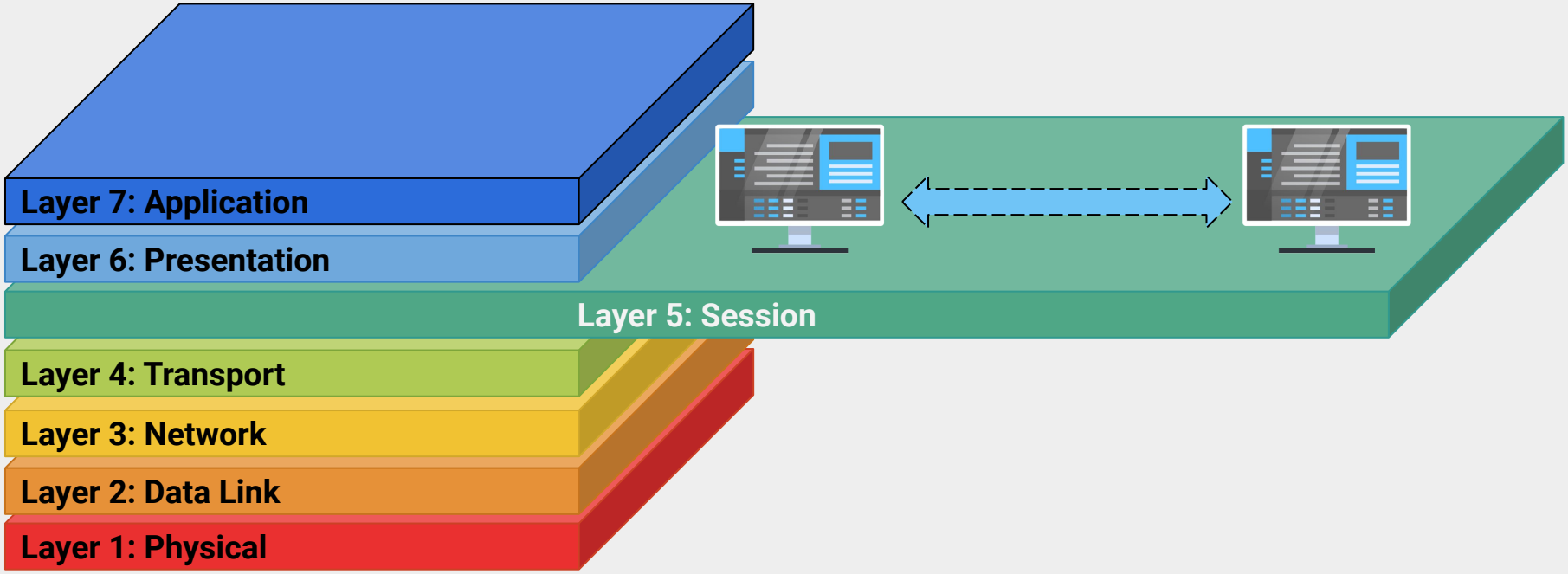
Layer 4: Transport

The **Transport layer** is responsible for actually transmitting data across the network. It puts data onto the network, and assigns source and destination ports.



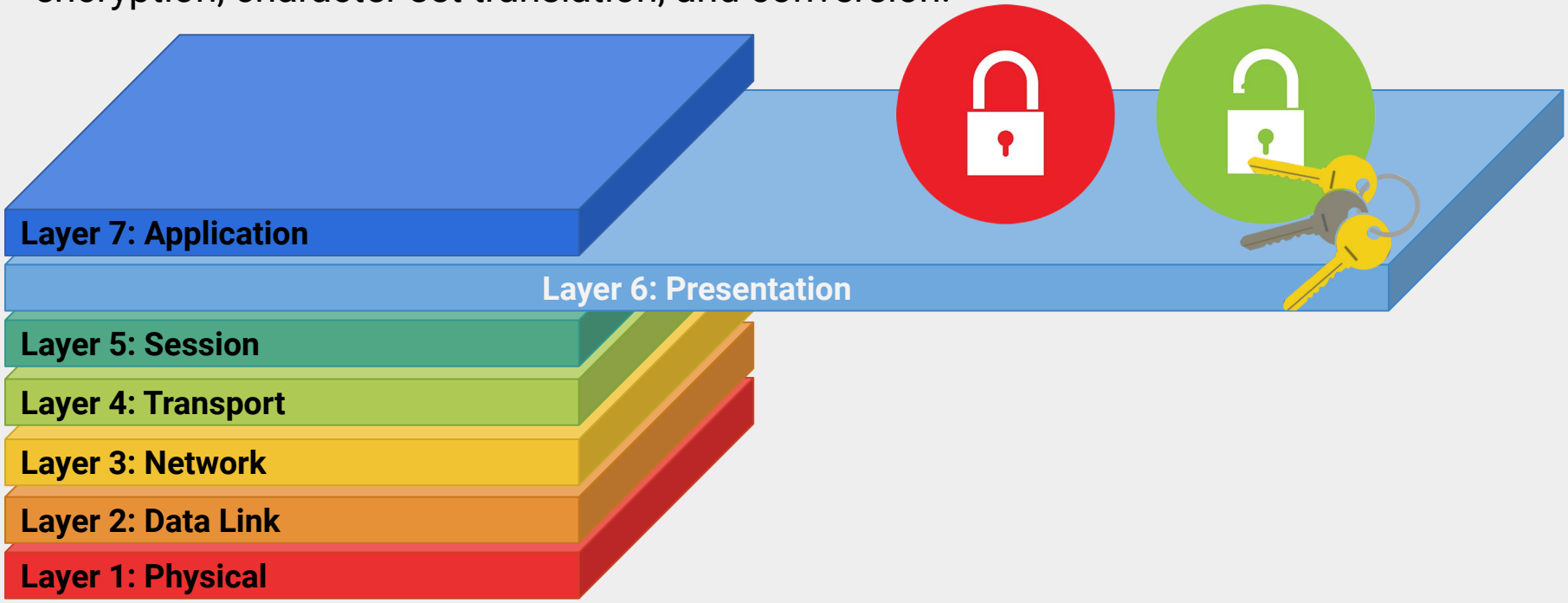
Layer 5: Session

The **Session layer** manages connections between ports on computers and handles data flow.



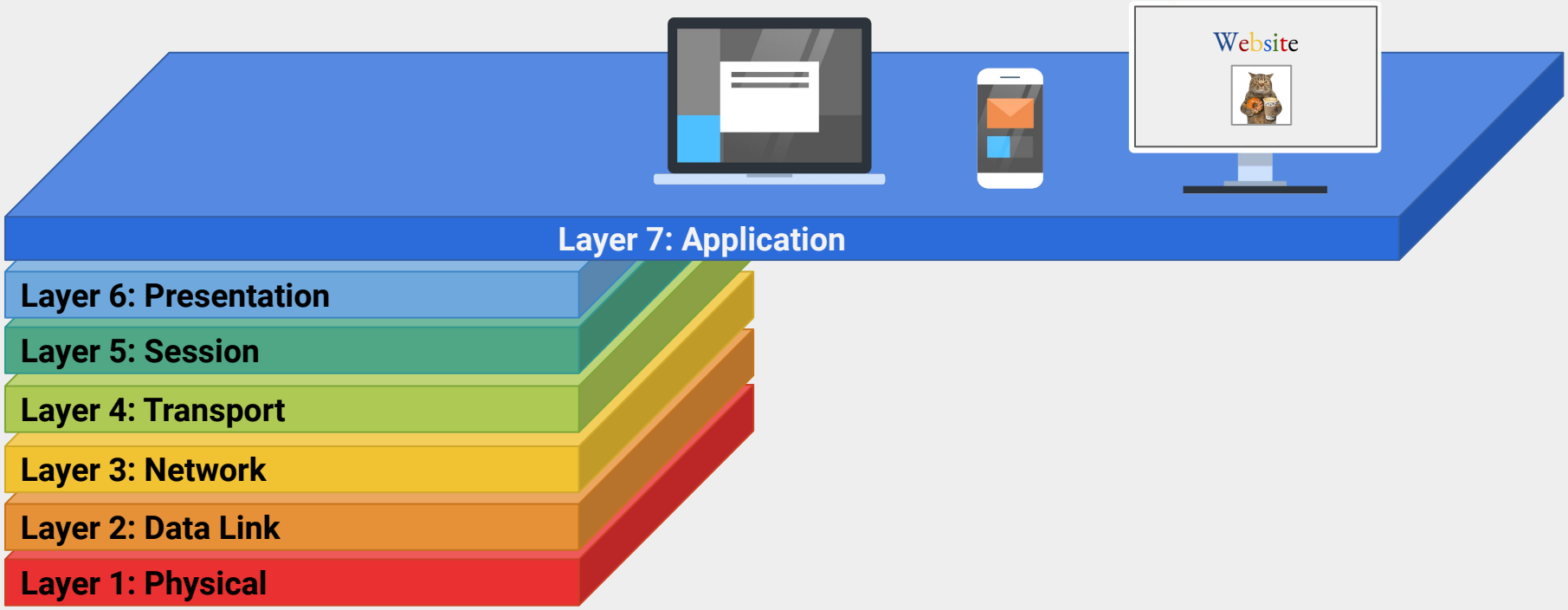
Layer 6: Presentation

The **Presentation layer** is is the translator for the network. It formats data to be presented to the Application layer, handles data representation, decryption and encryption, character set translation, and conversion.



Layer 7: Application

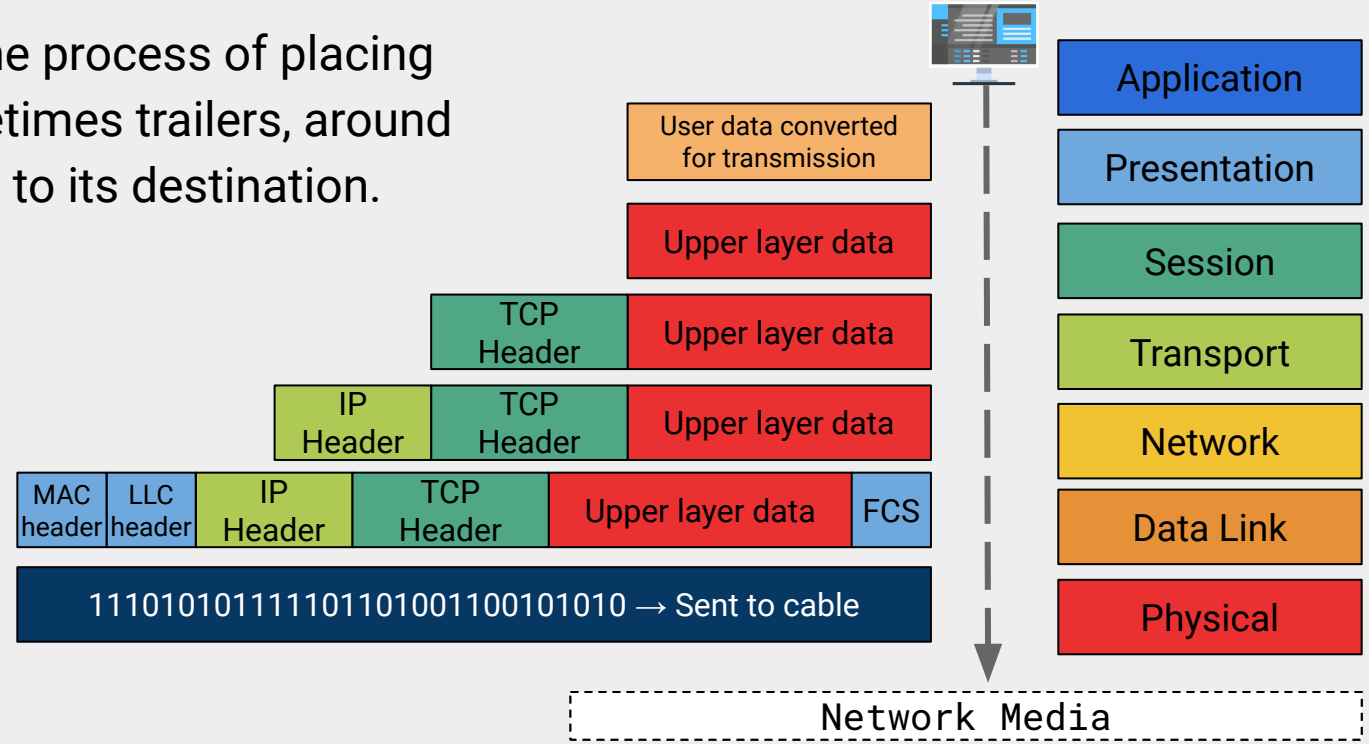
The **Application layer** represents data so the consuming application understands it. This is the layer an individual interacts with, such as a web or email application.



Encapsulation: Data Traveling Through Layers

Data moves through the layers, starting from Layer 7 and ending at Layer 1, in a process known as **encapsulation**.

Encapsulation is the process of placing headers, and sometimes trailers, around the data to direct it to its destination.

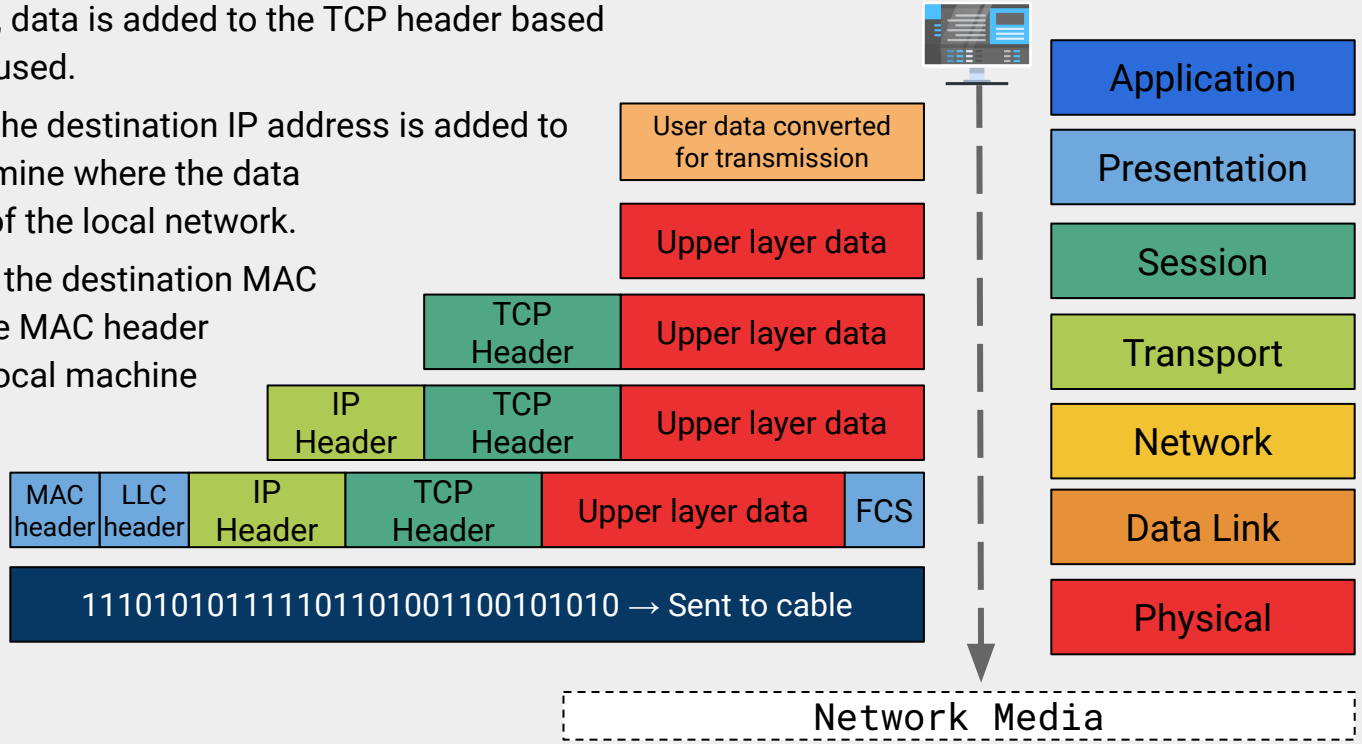


Encapsulation Example

These examples of encapsulation occur as data moves across layers:

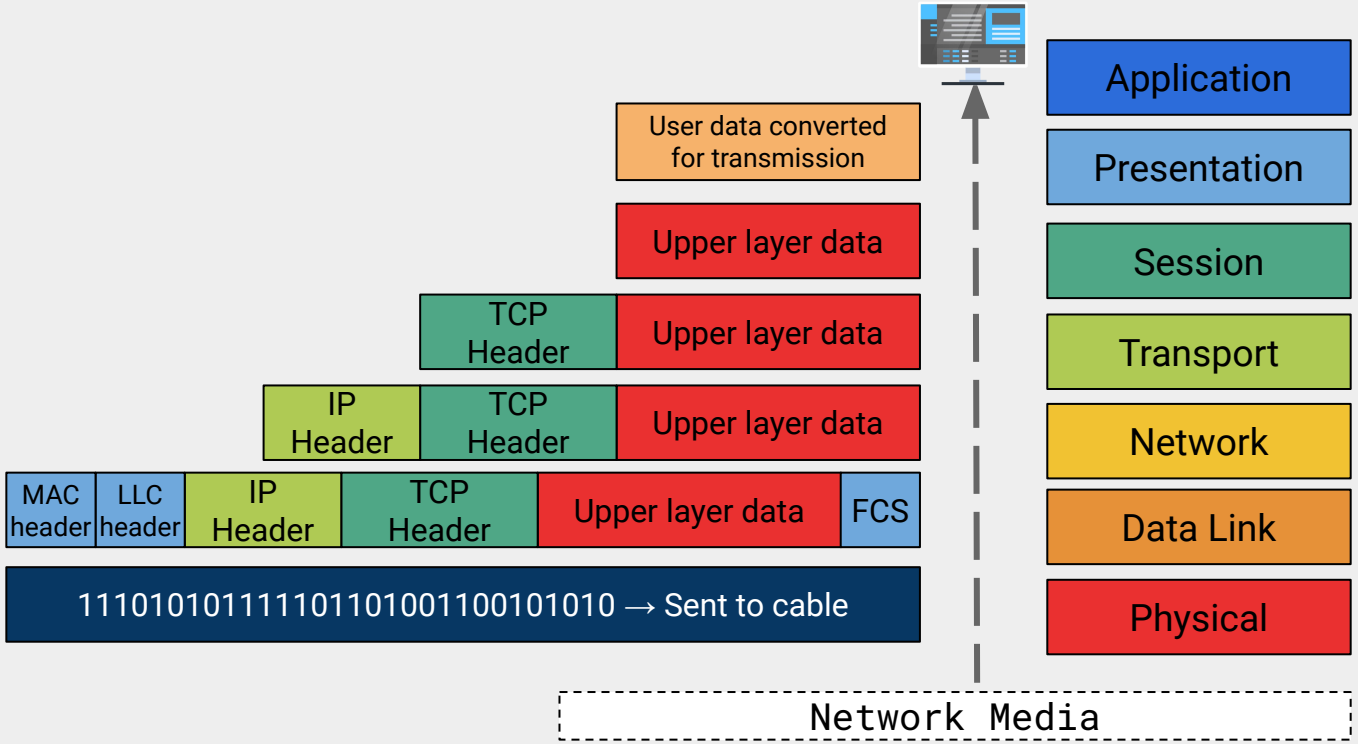
- At the **Transport layer**, data is added to the TCP header based on the protocol being used.
- At the **Network layer**, the destination IP address is added to the IP header to determine where the data is being sent outside of the local network.
- At the **Data Link layer**, the destination MAC address is added to the MAC header in to determine what local machine to send the data to.

In summary, TCP, IP, and MAC headers encapsulate the data.



Decapsulation

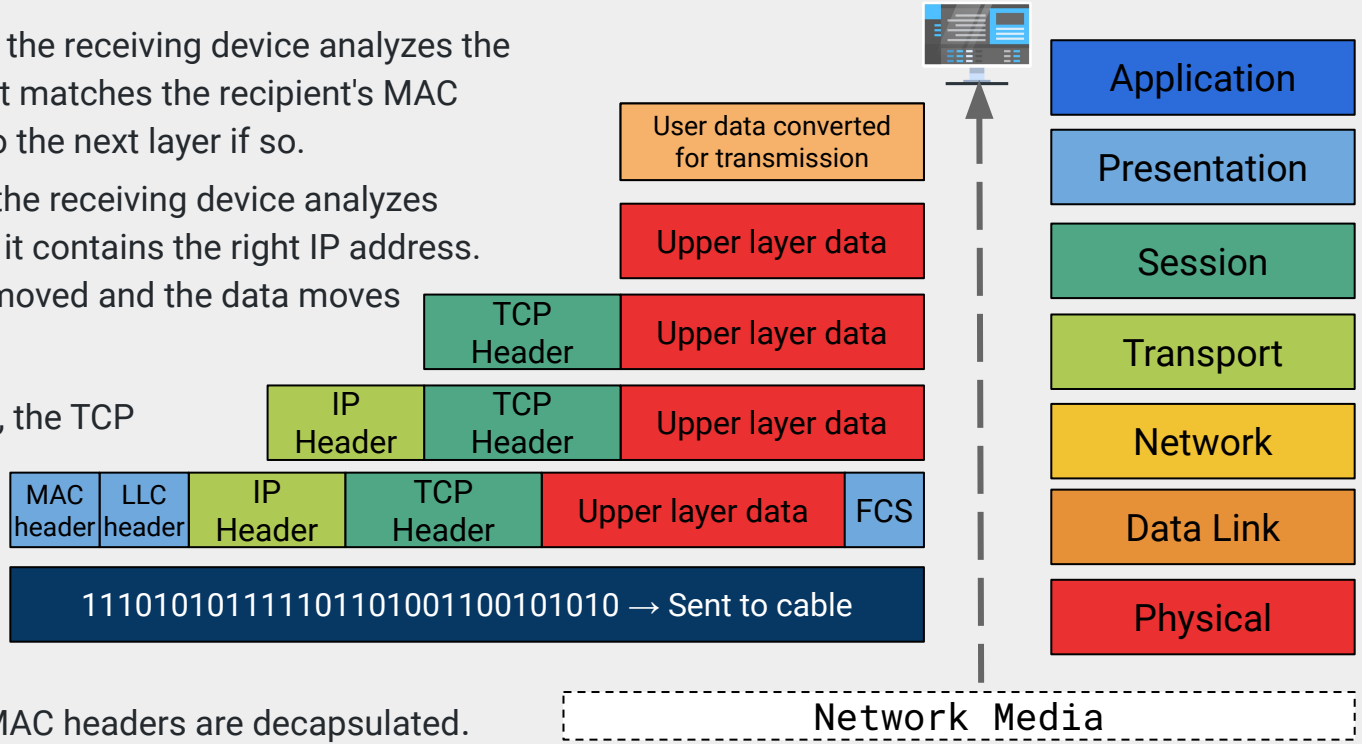
Decapsulation is the process of removing the headers, and sometimes trailers, around the data to confirm the data reaches the destination.



Decapsulation

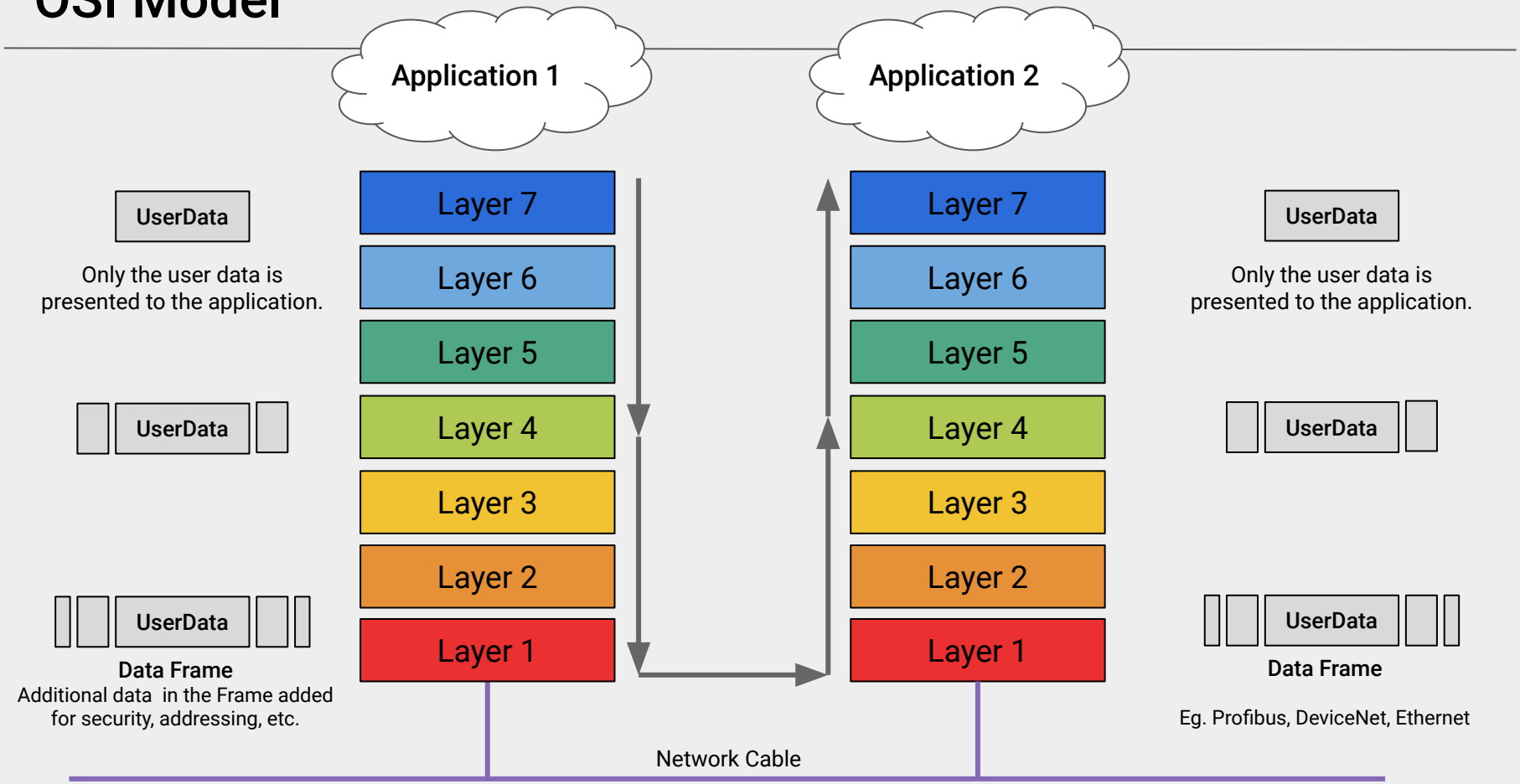
These examples of decapsulation occur as data moves across layers in reverse order:

- At the **Data Link layer**, the receiving device analyzes the MAC header to see if it matches the recipient's MAC address, and moves to the next layer if so.
- At the **Network layer**, the receiving device analyzes the IP header to see if it contains the right IP address. If so, the header is removed and the data moves to the next layer.
- At the **Transport layer**, the TCP header is analyzed to determine the destination port for further processing the data.



In summary, TCP, IP, and MAC headers are decapsulated.

OSI Model


Application 1
Application 2
UserData
Only the user data is presented to the application.
Layer 7
Layer 6
Layer 5
Layer 4
Layer 3
Layer 2
Layer 1
UserData
UserData
Only the user data is presented to the application.
UserData
UserData
Data Frame
Additional data in the Frame added for security, addressing, etc.
Data Frame
Eg. Profibus, DeviceNet, Ethernet
Network Cable

OSI Model

An email moving through the OSI layers:

- **Layer 7: Application**: The user types an email in Microsoft Outlook.
- **Layer 6: Presentation**: The email text is converted from plain text into a version the receiving server can understand.
- **Layer 5: Session**: A session with the receiving mail server is initiated.
- **Layer 4: Transport**: Since email uses SMTP, it assigns the SMTP destination port and initiates a handshake with the mail server.
- **Layer 3: Network**: The mail server destination IP is added.
- **Layer 2: Data Link**: The MAC address of the router is added so the email can be sent outside of the local network.
- **Layer 1: Physical:** The digital email is converted into a signal to be transmitted over a physical cable.

OSI and Security

The OSI model helps us more easily understand new protocols.

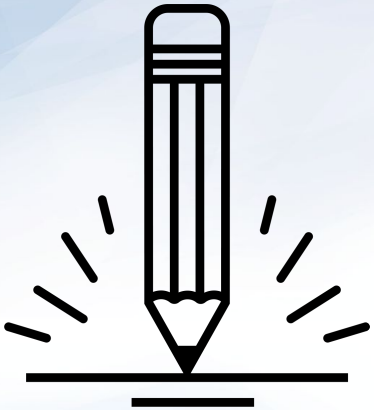
For example: If you find out that NetBIOS is a Layer 5 protocol, you immediately know that it's involved in managing user sessions, even if you've never heard of NetBIOS before.

The OSI model helps determine where problems in the network are occurring, even if we don't have full knowledge of the issue.

For example: If you realize you're having a Layer 3 issue, you know you should start investigating your routers, even if you don't know exactly what the problem is.

The OSI model makes it easier to communicate where a security attack has occurred and what should be done.

For example: If you know a SQL injection attack is occurring, you can explain to your management that you need a Layer 7 web application firewall to identify and mitigate the attack.



Activity: OSI Layers

In this activity, you will continue to play the role Security Analysts at Acme Corp.

You will be analyzing 10 suspicious network-related activities that have recently occurred at Acme Corp. Your task is to document at which OSI layer each of these situations occurred.

Suggested Time:
15 Minutes

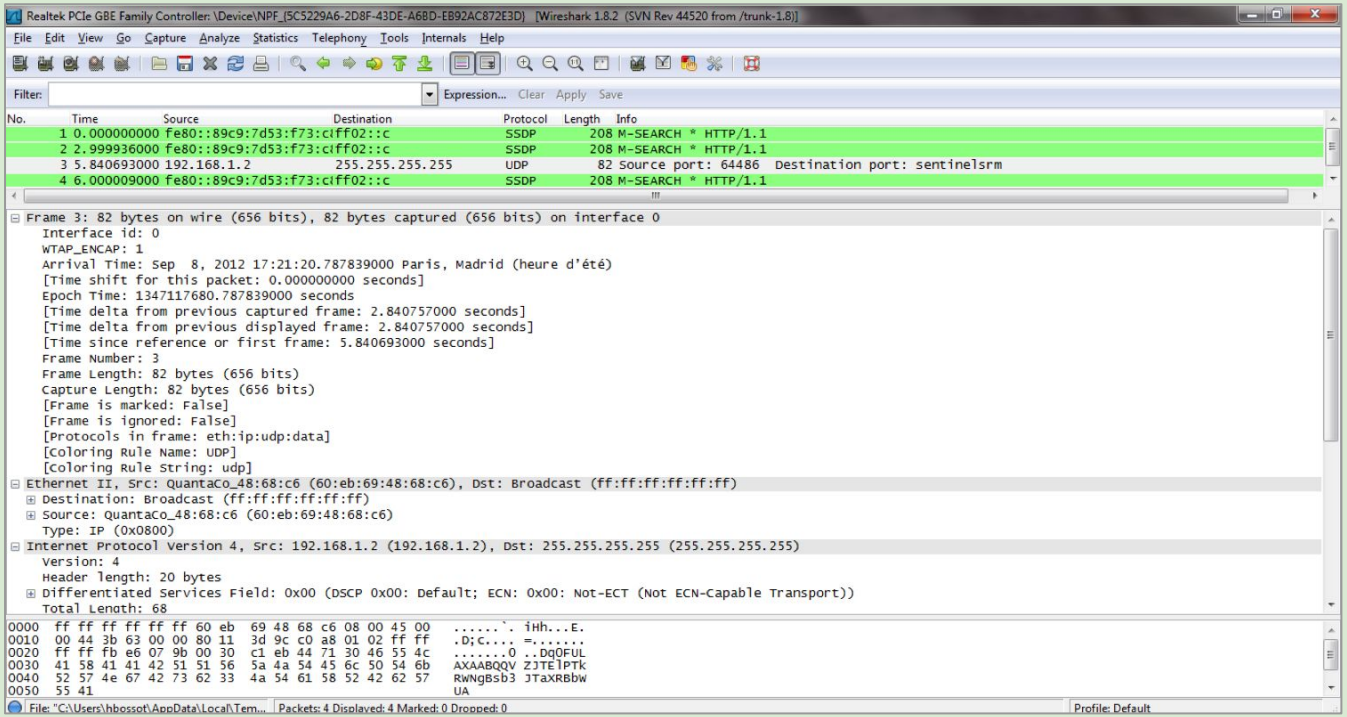
Time’s Up! Let’s Review.

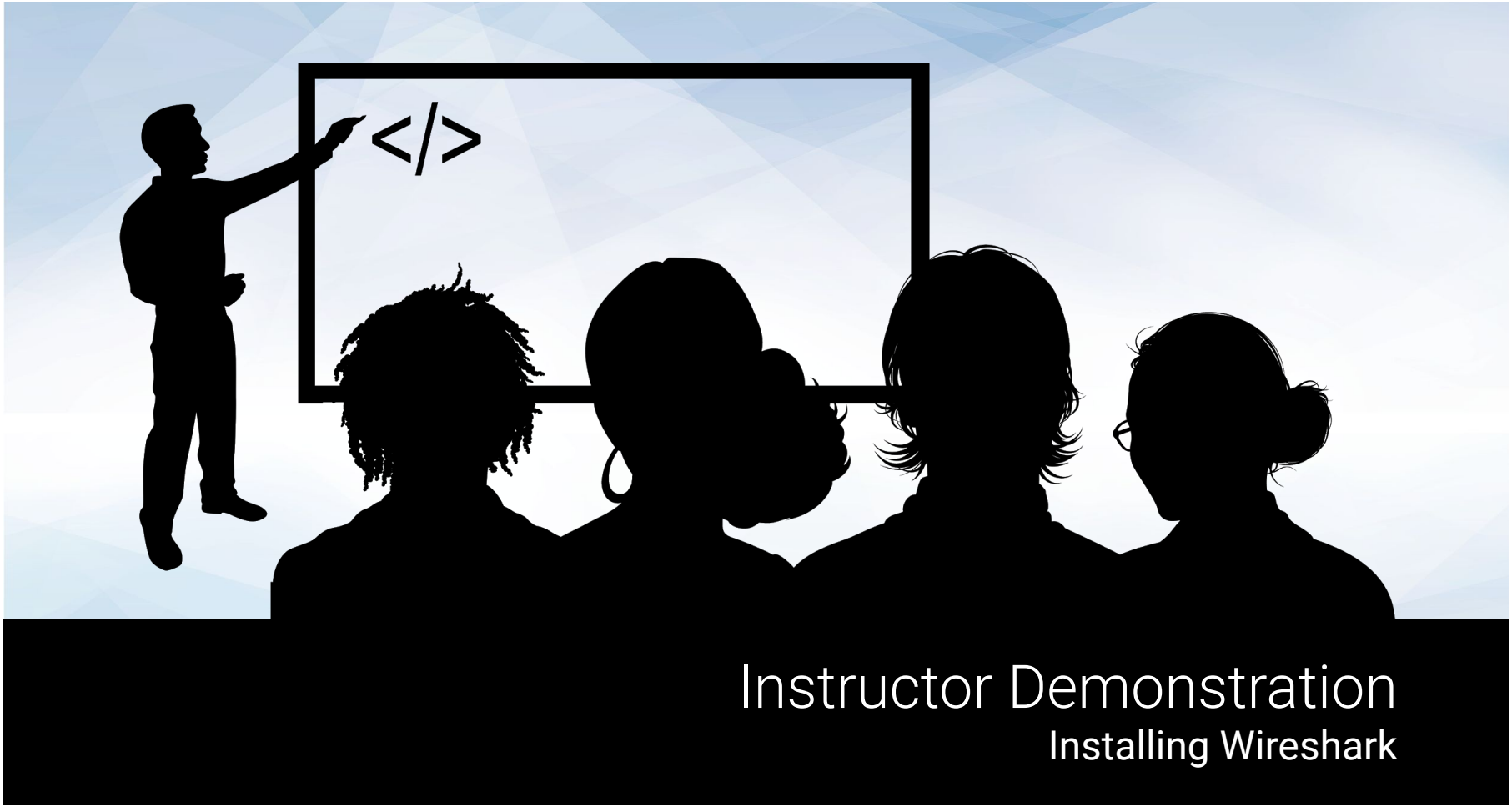
Wireshark

Wireshark is a tool that allows us to look at real-time communication across a network and monitor the activities of connected devices.

Packets and Wireshark

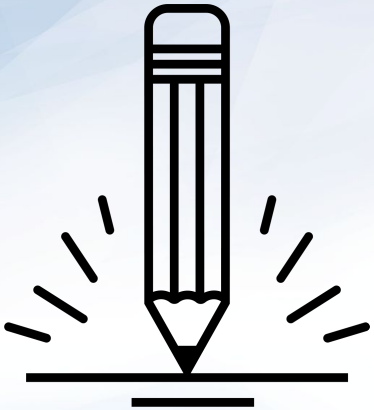
Communication between devices over a network is facilitated through the transfer of packets.





Instructor Demonstration

Installing Wireshark



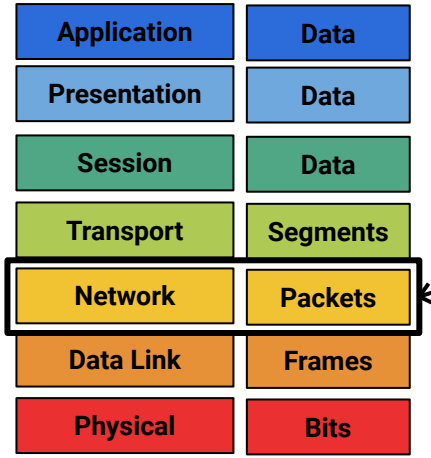
Activity: Wireshark Install

In this activity, you will be installing the Wireshark desktop application which will be used several upcoming activities.

Suggested Time:
10 Minutes

Revisiting Packets

Networks communicate with sequences of binary data called **packets**.
Wireshark is a **packet capturing tool**.



Header	Sender's IP address Receiver's IP address Protocol	96 bits
Payload	Data	660 bits
Trailer	Indicates end of packet, error correction	32 bits

Each packet contains fields of information such as the address of its origin, the destination address, and the information that connects to the related packets being sent.

Communicating over a network is not entirely secure. These packets can be intercepted and analyzed by other users on the network.

Packets and Wireshark

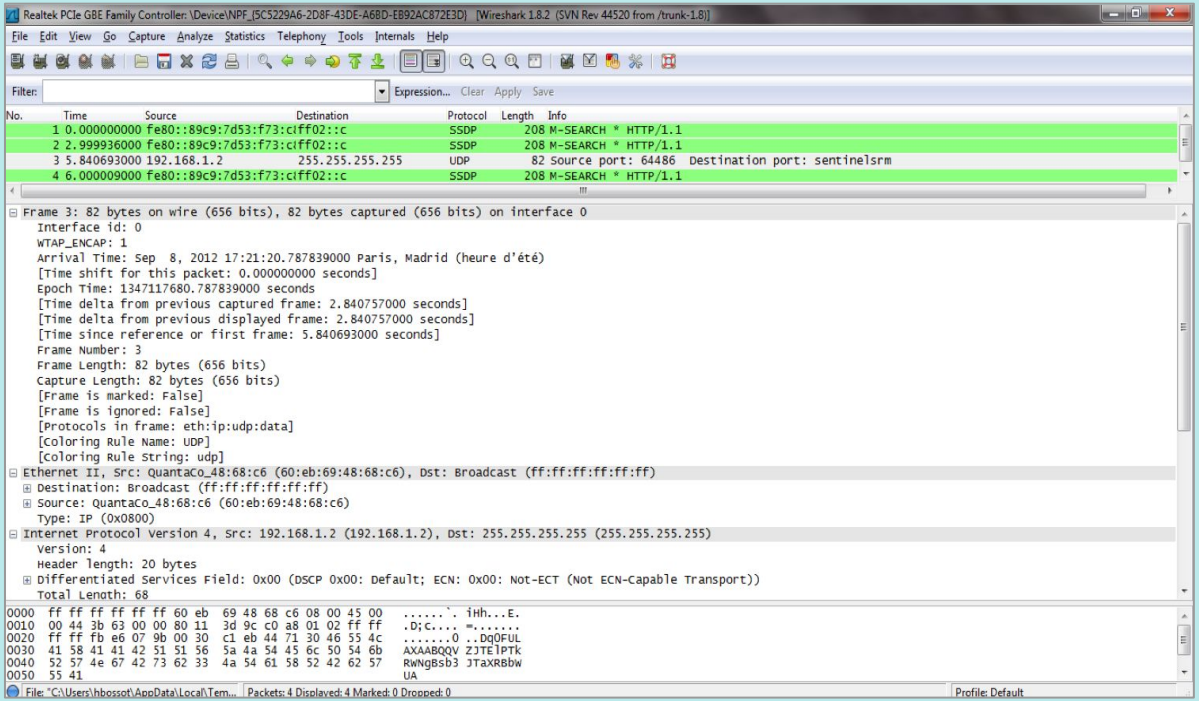
Wireshark is a tool that allows us to look at real-time communication across a network, and monitor the activities of the devices connected to it.

Wireshark does this analysis by inspecting individual packets.

Multiple packets collected into a file by Wireshark are called a **packet capture**. These have file extensions such as .cap, .pcap, and pcapng.

Packets and Wireshark

In these packet captures, Wireshark collects and analyzes the kinds of websites and webpages individuals on the network are viewing, as well as the type of communication occurring.



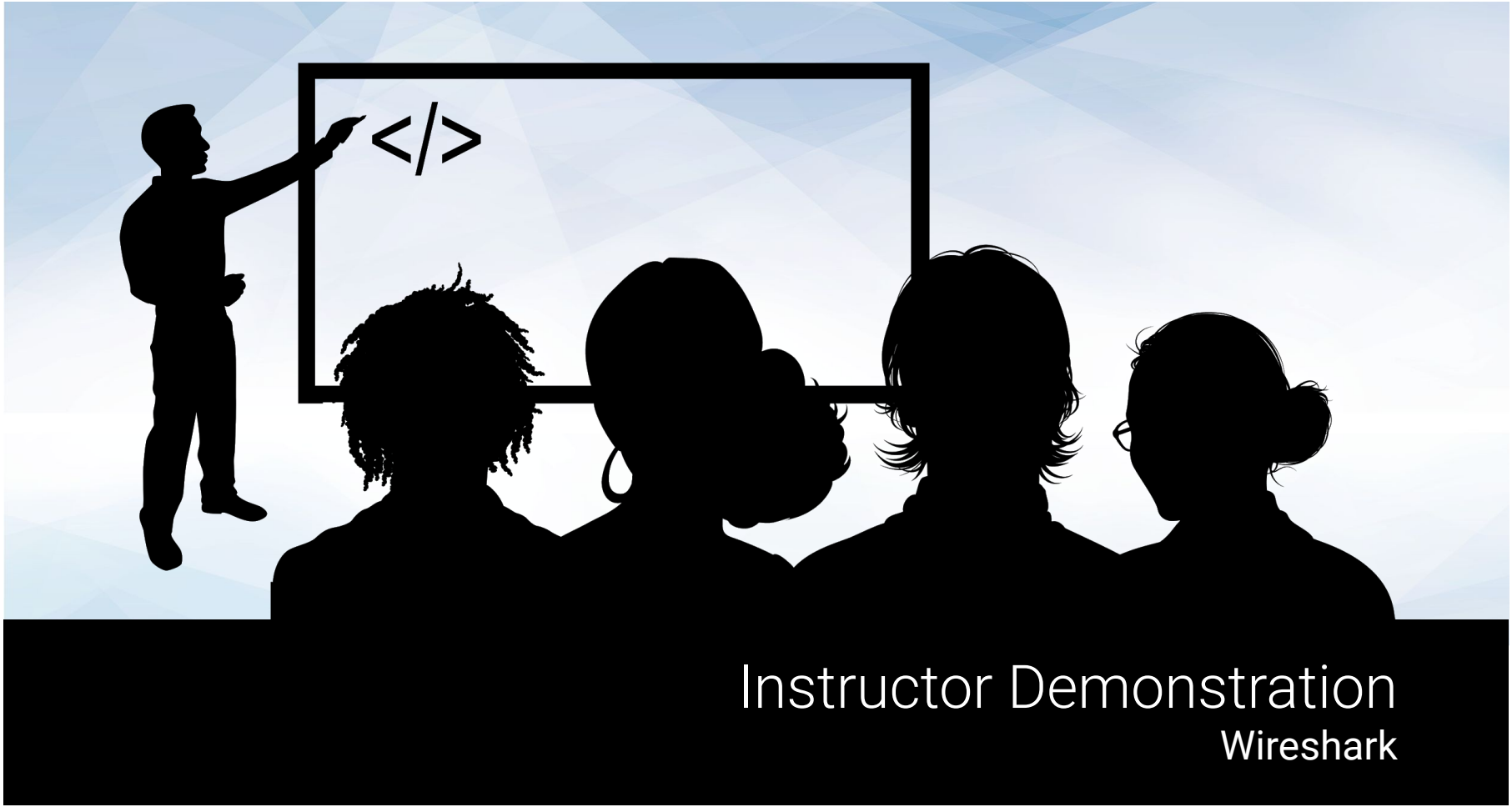
Wireshark Professional Context

Security professionals will often analyze network logs in order to research security-related issues.

For example:

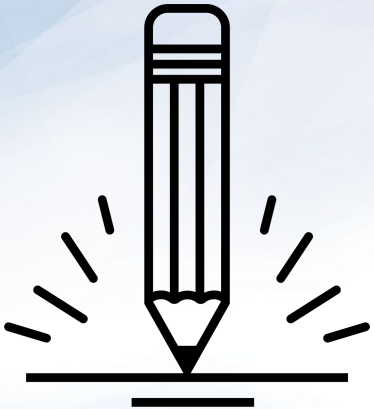
- Your manager has tasked you with analyzing web traffic to determine which source ports your system is using for HTTP requests. They want to make sure these aren't being blocked by your firewall.
- You've been provided a capture of the logs they want you to analyze.

We'll cover these scenarios in the next demo.



Instructor Demonstration

Wireshark



Activity: Capturing Packets

In this activity, you will continue to play the role of a security analyst at Acme Corp.

You will configure your Wireshark application with the five requested configuration settings provided by your manager.

Suggested Time:
15 Minutes

Time’s Up! Let’s Review.

Analyzing HTTP Traffic Setup

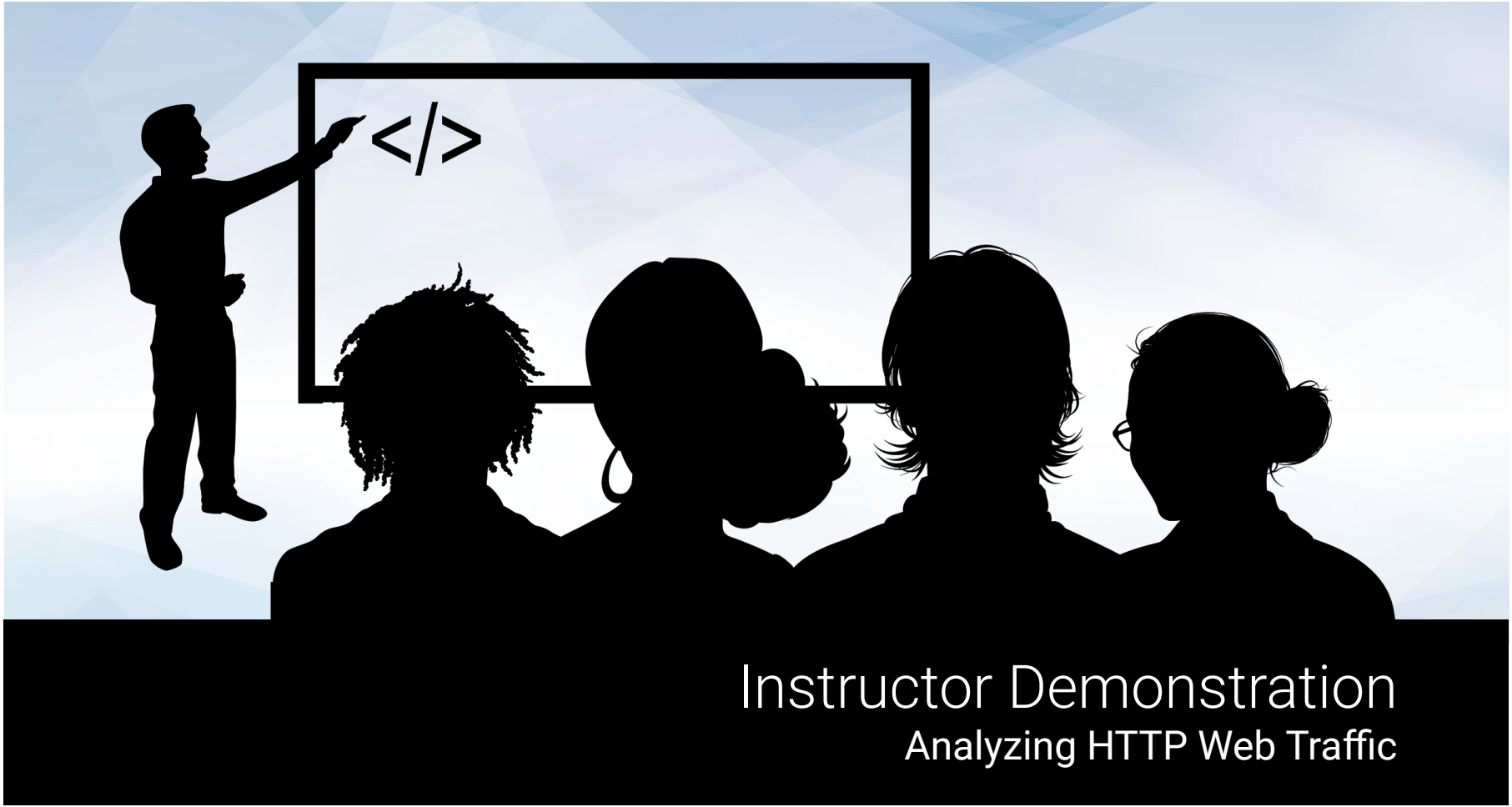
In the next demonstration, we will analyze HTTP web traffic with the following scenario:

Your manager wants you to make sure a new employee, Michael, is in fact working hard on his first day of work.

You could ask Michael if you could view his browser history, but, he could have cleared his browser history, or used more than one browser.

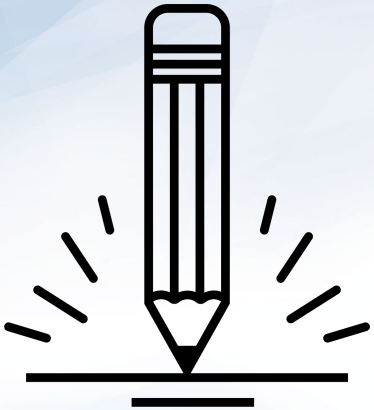
Fortunately, the networking team has a packet capture of Michael's web traffic, and you can use Wireshark to easily analyze the following:

- What websites has Michael visited?
- Were any communications sent from these websites?



Instructor Demonstration

Analyzing HTTP Web Traffic



Activity: Analyzing HTTP Data

In this activity, you will analyze web traffic to determine if Sally Stealer is a spy for your rival company, WidgetCorp.

You will also inspect the logs to see if Sally is sending any communications to WidgetCorp.

Suggested Time:
15 Minutes

Time’s Up! Let’s Review.

Questions?